IN THE CLAIMS:

Please AMEND claims 1, 8, 15, 17, 24, and 31, as shown below.

1. (Currently Amended) A method, comprising:

receiving a second data record to be stored on a single database;

retrieving a first integrity checksum stored with a first data record previous to the second data record, wherein the first data record and the second data are consecutive data records in the database;

computing a second integrity checksum for the second data record with a cryptographic method based on a storage key, the retrieved first integrity checksum and the second data record; and

storing the second data record and the second integrity checksum on the database.

2. (Previously Presented) The method according to claim 1, further comprising:

configuring the storage key to be a secret key of public key infrastructure.

3. (Previously Presented) The method according to claim 1, further comprising:

configuring the retrieved integrity checksum for a first row of the database to be a generated initialization vector.

4. (Previously Presented) The method according to claim 1, further comprising:

configuring the retrieved integrity checksum for a first row of the database to be a digital signature of a signing entity.

5. (Previously Presented) The method according to claim 1, wherein the retrieving the first integrity checksum comprises retrieving the first integrity checksum from a memory of a signing entity.

6. (Previously Presented) The method according to claim 1, further comprising:

storing the second integrity checksum on a memory of a signing entity.

7. (Previously Presented) The method according to claim 1, further comprising:

configuring the integrity checksums to comprise a running sequence number.

8. (Currently Amended) A method, comprising:

retrieving a second data record to be verified from a ~~first~~ single database;

retrieving a second integrity checksum of the second data record, wherein the first data record and the second data are consecutive data records in the database;

retrieving a first integrity checksum of a first data record previous to the retrieved second data record;

computing a third integrity checksum for the second data record based on the retrieved second data record, the first integrity checksum, and a storage key; and

comparing the second integrity checksum to the third integrity checksum, wherein the second data record is considered authentic when the second integrity checksum and the third integrity checksums are equal.

9. (Previously Presented) The method according to claim 8, further comprising:

configuring the storage key to be a public key of public key infrastructure.

10. (Previously Presented) The method according to claim 8, further comprising:

configuring the retrieved integrity checksum for a first row of the database to be a generated initialization vector.

11. (Previously Presented) The method according to claim 8, further comprising:

configuring the retrieved integrity checksum for a first row of the database to be a digital signatory of a signing authority.

12. (Previously Presented) The method according to claim 8, wherein the retrieving the first integrity checksum comprises retrieving the first integrity checksum from a memory of a verification entity.

13. (Previously Presented) The method according to claim 8, further comprising:

storing the second integrity checksum on a memory of a verification entity.

14. (Previously Presented) The method according to claim 8, further comprising:

configuring the integrity checksums to comprise a running sequence number.

15. (Currently Amended) A system, comprising:

a single database configured to store and provide signed data;

a data source configured to provide data records to be stored on the database ~~system~~;

a signing entity configured to sign data records to be stored on the database system with a second integrity checksum computed based on a second data record, a first integrity checksum of the first data record previous to the second data record to be signed, and a storage key, wherein the first data record and the second data are consecutive data records in the database; and

a verification entity configured to verify integrity of chosen data records by computing a computed third integrity checksum based on the second data record, the first integrity checksum of the first data record previous to the second data record, and the storage key, and comparing the computed third integrity checksum to the second integrity checksum stored on the database.

16. (Previously Presented) The system according to claim 15, wherein the signing entity and verification entity are configured to apply public key infrastructure to calculate and verify at least one of the first integrity checksum or the second integrity checksum.

17. (Currently Amended) A computer program embodied on a computer readable medium, said computer program for storing data records on a database system in which a signing entity is used for signing data records, wherein the computer program performs a process comprising the following, when executed in a computer device:

receiving a second data record to be stored on a single database;

retrieving a first integrity checksum stored with a first data record previous to the second data record, wherein the first data record and the second data are consecutive data records in the database;

computing a second integrity checksum for the second data record with a cryptographic method based on a storage key, the retrieved first integrity checksum and the second data record; and

storing the second data record and the second integrity checksum on the database.

18. (Previously Presented) The computer program according to claim 17, wherein the storage key is a secret key of public key infrastructure.

19. (Previously Presented) The computer program according to claim 17, wherein the retrieved integrity checksum for a first row of the database is a generated initialization vector.

20. (Previously Presented) The computer program according to claim 17, wherein the retrieved integrity checksum for a first row of the database is a digital signatory of the signing entity.

21. (Previously Presented) The computer program according to claim 17, wherein the first integrity checksum is retrieved from a memory of the signing entity.

22. (Previously Presented) The computer program according to claim 17, wherein the second integrity checksum is stored on a memory of the signing entity.

23. (Previously Presented) The computer program according to claim 17, wherein the integrity checksums comprise a running sequence number.

24. (Currently Amended) A computer program embodied a computer-readable medium for verifying the integrity of data records on a single database, wherein the computer program performs a process comprising the following, when executed in a computer device:

retrieving a second data record to be verified from a database;

retrieving a second integrity checksum of the second data record to be verified from a database;

retrieving a first integrity checksum of a first data record previous to the retrieved second data record, wherein the first data record and the second data are consecutive data records in the database;

computing a third integrity checksum for the second data record based on the retrieved second data record, the first integrity checksum, and a storage key; and

comparing the second integrity checksum to the third integrity checksum, wherein the second data record is considered authentic when the second integrity checksum and the third integrity checksums are equal.

25. (Previously Presented) The computer program according to claim 24, wherein a storage key is a public key of public key infrastructure.

26. (Previously Presented) The computer program according to claim 24, wherein the retrieved integrity checksum for a first row of the database is a generated initialization vector.

27. (Previously Presented) The computer program according to claim 24, wherein the retrieved integrity checksum for a first row of the database is a digital signatory of a signing authority.

28. (Previously Presented) The computer program according to claim 24, wherein the first integrity checksum is retrieved from a memory of a verification entity.

29. (Previously Presented) The computer program according to claim 24, wherein the second integrity checksum is stored on a memory of a verification entity.

30. (Previously Presented) The computer program according to claim 24, wherein the integrity checksums comprise a running sequence number.

31. (Currently Amended) A system, comprising:

storage means for storing and providing signed data, wherein the storage means is singular;

provision means for providing data records to be stored on the storage means;

signing means for signing data records to be stored on the storage means with a second integrity checksum computed based on a second data record, a first integrity checksum of the first data record previous to the second data record to be signed, and a storage key, wherein the first data record and the second data are consecutive data records in the database; and

verification means for verifying integrity of chosen data records by computing a computed third integrity checksum based on the second data record, the first integrity checksum of the first data record previous to the second data record, and the storage key, and comparing the computed third integrity checksum to the second integrity checksum stored on the storage means.

32. (Previously Presented) The system of claim 31, wherein the signing means and verification means are configured to apply public key means for calculating and verifying at least one of the first integrity checksum or the second integrity checksum.